

**WHAT IS CLAIMED IS:**

1. A memory system, comprising:

a hard disk, wherein the hard disk must be spun to be accessed;  
a cache memory, wherein the cache memory is comprised of non-volatile  
5 memory;

a memory controller, operable to:

determine if a memory request received by the memory system can be  
satisfied by accessing the cache memory;  
queue up memory requests if the memory request cannot be satisfied by  
10 the cache memory; and  
execute the memory requests queued up when the hard disk is accessed.

2. The system of claim 1, wherein the cache memory further comprises a polymer ferroelectric memory.

3. The system of claim 1, wherein the memory controller further comprises a digital  
15 signal processor.

4. The system of claim 1, wherein the memory controller further comprises an application specific integrated circuit.

5. The system of claim 1, wherein the memory controller further comprises software running on a host processor.

- 20 6. The system of claim 1, wherein the memory controller resides coincident with the cache memory.

7. The system of claim 1, wherein the memory controller resides separately from both the cache memory and the hard disk.

10. A method of processing memory requests, the method comprising:

- receiving a request for a memory operation;
- determining if data for the memory operation already exists in a cache memory;
- performing a cache memory operation, if the data already exists in the cache;
- if the data does not already exist in the cache:
  - accessing a hard disk that contains the data for the memory request;
  - performing a disk memory operation; and
  - performing any queued up disk memory operations.

11. The method of claim 10, wherein the memory operation is a read operation.

10 12. The method of claim 10, wherein accessing a hard disk further comprises spinning up the hard disk.

13. The method of claim 12, the method further comprising spinning down the hard disk after performing any queued up disk memory operations.

14. The method of claim 10, wherein if the data does not already exist in the cache,  
15 the method further comprising:

- determining if the request is part of a sequential stream;
- if request is part of a sequential stream, deallocating cache lines in the cache memory and prefetching new cache lines;
- if request is not part of a sequential stream, determine if prefetch is desirable; and
- if prefetch is desirable, prefetch data.

20 15. The method of claim 14, wherein the prefetch is queued up as a disk memory operation.

16. The method of claim 10, wherein performing any queued up disk memory operations further comprises determining if the queued up disk memory operations are  
25 desirable and then performing the queued up disk memory operations that are desirable.

17. The method of claim 10, wherein the memory operation is a write operation.
18. The method of claim 10, wherein the cache operation further comprises writing data into the cache.
19. The method of claim 18, wherein the cache operation further comprises queuing up a disk memory operation, wherein the disk memory operation will transfer the data to the disk.
20. The method of claim 19, wherein the queued up disk memory operations are periodically reviewed to ensure their continued desirability.
21. The method of claim 10, wherein the disk memory operation further comprises writing data to the disk.
22. The method of claim 10, wherein the queued up memory operations include writing data from the cache to the disk.

30. A method of performing a read memory operation, the method comprising:  
receiving a read request;  
determining if data to satisfy the read request is located in the cache;  
satisfying the read request from data in the cache, if the data is located in the  
cache;  
if the data is not located in the cache, performing a disk read operation, wherein  
the disk read operation comprises:  
accessing the disk;  
allocating a new cache line;  
transferring data from the disk to the new cache line; and  
satisfying the request.
31. The method of claim 30, wherein accessing the disk further comprises spinning up  
a hard disk.
32. The method of claim 31, wherein the method further comprises spinning down the  
hard disk after satisfying the request.
33. The method of claim 30, wherein the disk read operation further comprises:  
determining if the data transferred from the disk to the new cache line is part of a  
sequential stream;  
if the data is part of a sequential stream, prefetching new cache lines;  
if the data is not part of a sequential stream, determining if prefetch is desirable;  
and  
if prefetching is desirable, performing a prefetch.
34. The method of claim 30, wherein prefetching further comprises queuing up a  
prefetch operation to be executed during a next disk memory operation.

25

40. A method of performing a write memory request, the method comprising:

- receiving a write request;
- determining if at least one line in the cache is associated with the write request;
- if at least one line in the cache is associated with the write request, performing a

5 cache write to the line; and

- if no lines in the cache are associated with the write request, performing a new write operation.

41. The method of claim 40, wherein the new write operation further comprises:

- allocating a new cache line;
- writing data from the write request to the line allocated; and
- queueing up a disk write operation, wherein the disk write operation will transfer

10 the new data from the cache to a disk in a later disk memory operation.

- 42
50. An apparatus comprising:
- a storage device; and
  - a non-volatile cache memory coupled to the storage device.
51. The apparatus of claim 50 wherein the storage device includes a part capable of moving.
52. The apparatus of claim 51 further comprising:
- a controller coupled to the non-volatile cache memory to queue up input-output requests while the part is not moving.
53. The apparatus of claim 51 wherein the controller is adapted to perform the queued up input-output requests while the part is not moving.
54. The apparatus of claim 51 wherein the controller comprises software.
55. The apparatus of claim 54 wherein the apparatus further comprises a general-purpose processor coupled to the non-volatile cache memory, and the software comprises a driver for execution by the general-purpose processor.
56. The apparatus of claim 50 wherein the apparatus comprises a system selected from the group comprising a personal computer, a server, a workstation, a router, a switch, and a network appliance, a handheld computer, an instant messaging device, a pager and a mobile telephone.
57. The apparatus of claim 52 wherein the controller comprises a hardware controller device.
58. The apparatus of claim 50 wherein the storage device comprises a rotating storage device.
59. The apparatus of claim 58 wherein the rotating storage device comprises a hard disk drive.
60. The apparatus of claim 59 wherein the non-volatile cache memory comprises a polymer ferroelectric memory device.

61. The apparatus of claim 59 wherein the non-volatile cache memory comprises a volatile memory and a battery backup.

70. An apparatus comprising:

a rotating storage device;

a non-volatile cache memory coupled to the rotating storage device; and

a controller coupled to the cache memory and including:

5 means for queue first access requests directed to the rotating storage  
device;

means for spinning up the rotating storage device in response to second  
access requests; and

10 means for completing the queued first access requests after the rotating  
storage device is spun up.

71. The apparatus of claim 70 wherein the first access requests comprise write  
requests.

72. The apparatus of claim 71 wherein the second access requests comprise read  
requests.

15 73. The apparatus of claim 72 wherein the read requests comprise read requests for  
which there is a miss by the non-volatile cache memory.

74. The apparatus of claim 71 wherein the first access requests further comprise  
prefetches.

75. The apparatus of claim 74 wherein the read requests comprise read requests for  
20 which there is a miss by the non-volatile cache memory.

80. A method of operating a system which includes a rotating storage device, the method comprising:
- spinning down the rotating storage device;
  - receiving a first access request directed to the storage device;
  - 5 queuing up the first access request;
  - receiving a second access request directed to the storage device;
  - in response to receiving the second access request, spinning up the rotating storage device; and
  - servicing the second access request.
- 10 81. The method of claim 80 further comprising:  
servicing the first access request.
82. The method of claim 81 wherein the system further includes a cache coupled to the rotating storage device, and the second access request comprises a read request that misses the cache.
- 15 83. The method of claim 81 wherein the servicing of the first access request is performed after the servicing of the second access request.
84. The method of claim 83 wherein the second access request comprises a read request.
85. The method of claim 84 wherein the system further includes a cache, and the  
20 queuing up the first access request comprises recording the first access request in the cache.